



Analysis Prototype

Kevin Robair

24 August 1995

Analysis Prototype



- **Dynamic Linking and User Algorithms**
- **Expert Advisor/Decision Support System**
- **Numerical Analysis Tools**

Analysis Prototype



Dynamic Linking of User Algorithms

- **Purpose**
 - Establish reliability of dynamic linking as an implementation of user defined algorithms.
 - Verify capabilities of vendor supplied dynamic linking libraries provided with development systems.
- **Approach**
 - Used Sun libraries provided with Solaris 2.3 C++ compiler to build a dynamic linking class, which uses shared objects as input.
 - Used Sun Solaris 2.3 C, C++, and FORTRAN compilers to generate shared objects.
 - Shared objects were linked and executed using a prototype dynamic linking class.

Analysis Prototype



- **Results**
 - **Sun Solaris 2.3 proved to have excellent support for dynamic linking.**
 - **HP documentation promises similar support.**
 - **DEC, IBM, and SGI need further investigation, preliminary findings show support for dynamic linking among all three.**
 - **Dynamic linking of FORTRAN shared objects from a C++ application was NOT successful**
- **Impact on Design**
 - **Dynamic Linking is possible without vendor support, but is easier when support such as that provided with Solaris 2.3 is available**
- **Future Goals**
 - **Verify dynamic linking support on remaining platforms.**
 - **Explore alternative methods of using FORTRAN in dynamic user algorithms**

Analysis Prototype



Expert Advisor / Decision Support System

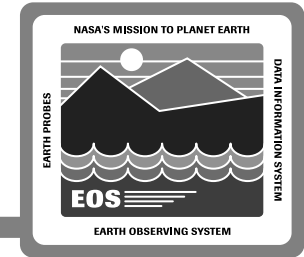
- **Goals**
 - Gain experience with RTworks
 - Establish data interface with RTworks
 - Transfer SSR model from previous PRR, and improve based on latest SSR information
 - Allow FOT to use RTworks, in order to verify ops concept of the Expert Advisor as a modifiable tool
 - Integrate C language function calls into RTworks rules
- **Approach**
 - Connected mock data source to RTworks
 - Incorporated basic SSR maintenance goals into a set of rules in RTworks
 - Allowed FOT members to modify rules
 - Created mock display to verify operation

Analysis Prototype



- **Results**
 - RTworks performed baseline SSR analysis smoothly
 - Data interface with RTworks was easily implemented
 - FOT found the RTworks rule editor easy to use, despite it's many features
 - C function calls were integrated into RTworks rules with minimal effort
- **Impact on Design**
 - FOS will use RTworks as the engine for the Expert Advisor and Solid State Recorder analysis tool
- **Future Goals**
 - Create a larger rule base to test RTwork's efficiency
 - Add more parameters, such that real-time telemetry data rates are input into RTworks
 - Use prototype to further explore SSR management approach

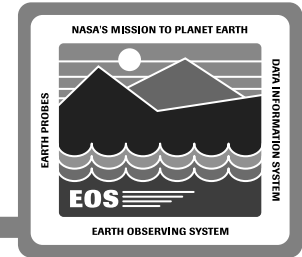
Analysis Prototype



Numerical Analysis Tools Study

- **Purpose**
 - Evaluate ECS available COTS packages for performing numerical analysis
 - IDL
 - IMSL
- **Approach**
 - Compare packages for baseline functionality
 - Compare packages for future extensibility
 - Compare packages for ease of use

Analysis Prototype



- **Results**

- **IDL**

- Requires connection to stand alone process**

- Interface to stand alone process is non-trivial**

- Error handling is difficult**

- Actual functions are very user friendly**

- **IMSL**

- C Library of numerical functions**

- Error handling is handed in standard C fashion**

- Functions are more complicated than IDL's and require more knowledge of complex numerical methods**

- **IMSL and IDL have similar functionality, however each package has features the other does not**

Analysis Prototype



- **Impact on Design**
 - **Since IMSL satisfies baseline requirements, it will be used to satisfy current requirements**
 - **IDL will still be available if added functionality is required but not supported by IMSL**